Discovery of Sets of Mutually Orthogonal Vanishing Points in Videos

Till Kroeger[1] Dengxin Dai[1] Radu Timofte[1] Luc Van Gool[1,2]
[1]Computer Vision Laboratory, D-ITET, ETH Zurich
[2]VISICS / iMinds, ESAT, K.U. Leuven
{kroegert, dai, timofter, vangool}@vision.ee.ethz.ch

Abstract

While vanishing point (VP) estimation has received extensive attention, most approaches focus on static images or perform detection and tracking separately. In this paper, we focus on man-made environments and propose a novel method for detecting and tracking groups of mutually orthogonal vanishing points (MOVP)*, also known as Manhattan frames, jointly from monocular videos. The method is unique in that it is designed to enforce orthogonality in groups of VPs, temporal consistency of each individual MOVP, and orientation consistency of all putative MOVP. To this end, the method consists of three steps: 1) proposal of MOVP candidates by directly incorporating mutual orthogonality; 2) extracting consistent tracks of MOVPs by minimizing the flow cost over a network where nodes are putative MOVPs and edges are putative links across time; and 3) refinement of all MOVPs by enforcing consistency between lines, their identified vanishing directions and consistency of global camera orientation. The method is evaluated on six newly collected and annotated videos of urban scenes. Extensive experiments show that the method outperforms greedy MOVP tracking method considerably. In addition, we also test the method for camera orientation estimation and show that it obtains very promising results on a challenging street-view dataset.*

1. Introduction

Often a number of simplifying assumptions are made in order to facilitate the reasoning about complex man-made environments. Most man-made structures can be described in terms of geometric primitives, such as parallel or orthogonal planes and lines. Under a projective transformation, sets of parallel lines often converge to an intersection point in the imaged scene. This point is known as a vanishing point (VP). The vanishing points provide strong cues for the 3D geometry of the scene. Since for scenes like urban environments the orthogonal planes are the dominant geometric primitives, one can constrain the detection to *mutually orthogonal vanishing points* (MOVP, also know as Manhattan frames [28]). One MOVP is depicted in Fig. 1. Generally in man-made environments, there can be multiple MOVPs present, which may or may not share one common VP. Often a clearly dominant MOVP is not present, as visualized in Fig. 7, and a set of MOVPs have to be estimated.

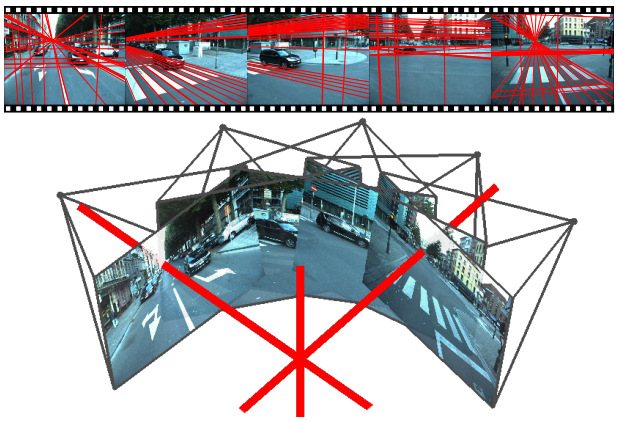

Figure 1: One *mutually orthogonal vanishing point* (MOVP) discovered from a video sequence and visualized using 5 frames. The discovered MOVP allows extraction of the global camera orientation for each frame.

Camera calibration [11], pose estimation [22], 3D reconstruction [7, 14], and autonomous navigation [23], are areas in the field of computer vision, where the VPs are used as low-level input. Many such applications, working on video sequences or image sets, require VP estimates in every frame and links across views or frames. If this is needed, the camera pose for each frame is usually assumed to be known [1, 12], facilitating the VP association across images, or the VP detection and tracking tasks are separated (greedy assignment [9] and particle filters [23, 25] are used). However, the pose knowledge is often not available, requires odometry or external motion measurements.

We propose the discovery of sets of MOVPs from videos where only the internal camera calibration is known. For

this purpose, the method is designed to leverage mainly three sources of information: orthogonality in groups of VPs, temporal consistency of each individual MOVP, and orientation consistency of all putative MOVPs. We extract MOVP proposals in each video frame by directly incorporating mutual orthogonality, then enforce temporal consistency by using a multi-target tracking formulation, and finally refine the MOVP tracks by enforcing consistency between lines and their identified MOVP and consistency of global orientation of all MOVP. Our main contributions are:

1. We are the first to consider the problem of discovery of multiple MOVPs from videos with unknown camera pose. We provide a new evaluation dataset for this task.

2. We adapt the established Multi-Target Tracking formulation using min-cost network flows to the problem of MOVP discovery.

3. We propose a Non-Linear Least-Squares refinement step to jointly refine all discovered MOVPs and to reliably extract the global camera orientation.

The method is tested on six newly collected videos of real urban scenes, in which all vanishing points are manually labeled. Extensive experiments validate the effectiveness of the method, especially for challenging scenes where multiple MOVPs, with equally strong line support, appear. Furthermore, we apply the method to the task of global camera orientation estimation and show promising results on the large, challenging Antwerp street-view dataset [16].

2. Related Work

VP extraction is a popular topic in computer vision. We categorize according to algorithmic design choices the most relevant recent literature:

Input: Most works start from lines [6, 9], or line segments [3, 21, 26, 11, 29, 12, 34, 2]. Some approaches employ continuous image gradients or texture [27, 25, 23] and thresholded edges images [30]. When the 3D geometry is known, the surface normals can be directly used [28].

Accumulator space: The intersections of imaged lines are computed in the (unbounded) image space [26, 27, 29, 9, 2, 35] or on a (bounded) Gaussian unit sphere, as introduced in [3] and used in [21, 24, 20, 1, 15, 11, 12, 4, 22, 28].

Line-VP consistency and VP refinement: The consistency between an estimated VP and the image lines is usually measured using line endpoint distances in the image, used by us and [29, 12, 4, 2, 17], the angular differences in the image [26, 8], with explicit probabilistic modeling of the line end point errors [35], or with angles between normals of interpretation planes in the Gaussian sphere [21, 24, 20]. We sample MOVP candidates on the Gaussian sphere because testing for orthogonality directly translates to vector cross products, but revert to image-space fitting errors for refinement to avoid distorted errors and to attenuate dependance on (potentially) noisy internal camera calibration.

The VP computation or refinement with given associated lines is commonly done by Hough voting and non-maximum suppression [21, 24, 20, 32, 23], by solving a quadratic program [2], implicitly in an Expectation-Maximization (EM) setting [1, 27, 29, 35], or by linear least-squares, as in [15].

Solution: For a final solution, different methods combine the input, the accumulator space and the line-VP consistency measures and refinement. Efficient search [26, 8], direct clustering [29, 17], multi-line RANSAC [4, 34], EM procedures [1, 15, 27, 12, 35], or MCMC inference [28] are among the methods employed directly on the accumulator space. If a discretization is enforced on the accumalator space, the solutions are found by voting schemes [21, 24, 20, 23] or inference over graphical models [30, 2].

Camera calibration and VP orthogonality: The internal camera calibration is assumed known in [24, 20, 25, 12, 9, 23], while others do not [21, 15, 26, 35, 2]. From the extracted VPs the internal parameters can be estimated [6, 11, 34]. Also VPs have been used for the estimation of external camera parameters such as the orientation of camera to scene [15] and the orientation of 3D shape to camera [3], and as additional constraints for full camera pose [22]. Often, the VPs are extracted by imposing further scene-dependent constraints. It is the case of mutual VP orthogonality constraint (or Manhattan World) [4, 8, 12, 9, 34], sets with a shared vertical VP (Atlanta World) [27, 2], and as in our paper, sets of mutually orthogonal VPs (MOVPs) [28].

Multi-view extraction and VP Tracking: [1] uses known camera poses to solve multi-view VP extraction by Hough voting with EM refinement. [12] uses Structure-from-Motion (SfM) camera pose estimates to extract orthogonal VPs independently in multiple views and enforce consistency. [9] extracts orthogonal VPs separately in each video frame. VP sets are then greedily linked across frames. [25, 23] aim at road direction finding based on tracked VPs. A single finite VP is extracted and tracked using particle filters. It corresponds to the heading direction.

3. Our Approach

We aim at the discovery of multiple sets of MOVPs. For this, we start from line segments as image primitives, The lines in the image space are the observations we make over the scene world and support the presence of MOVPs. Therefore, the sets of MOVPs compete on the set of observations. Since we work over a video sequence, the temporal consistency of the MOVPs is another key information we use. We expect that over a whole sequence a reduced set of MOVPs is capable to explain all the observations and to be

temporally consistent. Since all VPs are constant in space, and only the camera can move freely, temporal constancy of MOVPs directly translates to finding the global camera orientation in all frames, such that all locally extracted MOVPs are constant when transferred to the global reference frame.

In the following we derive the algorithmic formulation of our method. We will describe MOVP candidate generation in § 3.1, temporal linking in § 3.2 and refinement in § 3.3.

3.1. MOVP candidate extraction

For reasoning over VPs we first extract line segments [33] as image primitives. Since exhaustively searching for all line convergence points is intractable due to the large amount of line segments, we employ a 3-line RANSAC sampling to extract MOVP candidates [4]. In highly textured scenes the number of line segments is large, and in consequence we will obtain many duplicate MOVPs. We reduce all samples to a set of representative candidates in a subsequent non-maximum suppression step.

Additionally, we need an approximate orientation change $D_{n,n+1}$ between all pairs of frames n to $n+1$ in order to compute the linking cost between two MOVPs in § 3.2.2. This can be done using image descriptors, such as SIFT [19], feature matching, Essential matrix computation, and decomposition. SIFT features can be expensive to compute and match, and the orientation estimate may be noisy, as shown in the experiments in § 4.2. Because of this we chose to estimate the orientation change differently using a RANSAC process again: We randomly sample one MOVP candidate from frame n as well as $n+1$, compute the necessary camera orientation change for a perfect overlap, and compute the inliers, i.e. how many MOVP candidates from frame n find a close fit in frame $n+1$. For each frame n we keep the best orientation change $D_{n,n+1}$, which produces the most MOVP candidate inliers in the next frame.

3.2. Multi-MOVPs Tracking

The data association of the MOVPs extracted in each frame to global identities is formulated as a Maximum A Posteriori (MAP) problem. We follow (including the notations) the traditional approach of Zhang *et al*. [36] as used for multi-object tracking. We use a cost-flow network to model the problem and a min-cost flow algorithm to solve it. The intuition is that finding non-overlapping MOVPs tracks is analogous to finding edge-disjoint paths in a graph, which admits a solution by efficient network flow algorithms.

Let $\mathcal{X} = \{\mathbf{x}_i\}$ be the set of MOVP observations, each defined by a 3×3 orthonormal matrix $x_i \in \mathrm{SO}(3)$, and time step (frame index), $\mathbf{x}_i = (x_i, t_i)$. A time ordered list of MOVP observations represents a single track hypothesis, *i.e.* $T_k = \{\mathbf{x}_{k_1}, \mathbf{x}_{k_2}, \cdots, \mathbf{x}_{k_{l_k}}\}$ where $\mathbf{x}_{k_i} \in \mathcal{X}$ and l_k is the length. A set of such track hypotheses defines an association hypothesis, $\mathcal{T} = \{T_k\}$.

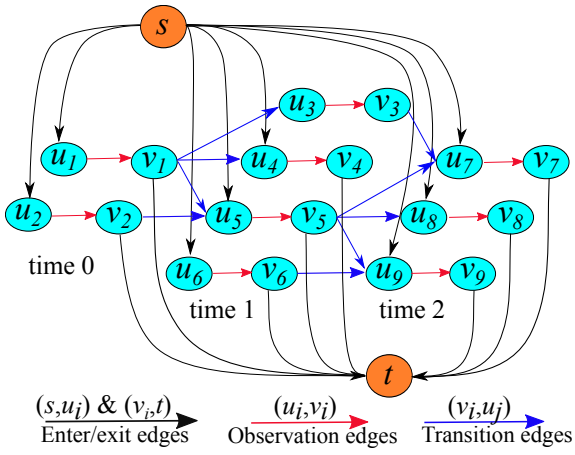

Figure 2: An example of the cost-flow network with 3 time steps and 9 observations (as in [36]).

The objective is to maximize the posteriori probability of $\mathcal{T}$ given the observation set $\mathcal{X}$:

$$\begin{aligned} \mathcal{T}^* &= \arg\max_{\mathcal{T}} P(\mathcal{T}|\mathcal{X}) \\ &= \arg\max_{\mathcal{T}} P(\mathcal{X}|\mathcal{T})P(\mathcal{T}) \\ &= \arg\max_{\mathcal{T}} \textstyle\prod_i P(\mathbf{x}_i|\mathcal{T})P(\mathcal{T}) \end{aligned} \tag{1}$$

under the assumption of conditional independence of the likelihood probabilities given the hypothesis $\mathcal{T}$.

Optimizing directly over the space of $\mathcal{T}$ is infeasible. The search space can be reduced if we use the fact that an observation can not belong to more than one track, therefore $T_k \in \mathcal{T}$ can not overlap with each other:

$$T_k \cap T_l = \emptyset, \forall k \neq l \tag{2}$$

Generally, MOVP tracks may not be independent. But if we assume the camera orientation to be given or computed (in our case by RANSAC in § 3.1), then we can assume independence of MOVP tracks. Thus, Eq. (1) becomes:

$$\begin{aligned} \mathcal{T}^* &= \arg\max_{\mathcal{T}} \textstyle\prod_i P(\mathbf{x}_i|\mathcal{T}) \prod_{T_k \in \mathcal{T}} P(T_k) \\ &\text{s.t.} \quad T_k \cap T_l = \emptyset, \forall k \neq l \end{aligned} \tag{3}$$

where

$$P(\mathbf{x}_i|\mathcal{T}) = \begin{cases} 1-\beta_i & \exists T_k \in \mathcal{T}, \mathbf{x}_i \in T_k \\ \beta_i & \text{otherwise} \end{cases} \tag{4}$$

$$\begin{aligned} P(T_k) &= P(\{\mathbf{x}_{k_0}, \mathbf{x}_{k_1}, \cdots, \mathbf{x}_{k_{l_k}}\}) \\ &= P_{entr}(\mathbf{x}_{k_0}) P_{link}(\mathbf{x}_{k_1}|\mathbf{x}_{k_0}), P_{link}(\mathbf{x}_{k_2}|\mathbf{x}_{k_1}) \\ &\quad \cdots P_{link}(\mathbf{x}_{k_{l_k}}|\mathbf{x}_{k_{l_k-1}}) P_{exit}(\mathbf{x}_{k_{l_k}}) \end{aligned} \tag{5}$$

$P(\mathbf{x}_i|\mathcal{T})$ is the likelihood for an observation $\mathbf{x}_i$, β_i being the false alarm probability of $\mathbf{x}_i$. $P(T_k)$ is the likelihood for

a track T_k and is modeled through a Markov chain of transition probabilities $P_{link}(\mathbf{x}_{k_{i+1}}|\mathbf{x}_{k_i})$, initialization P_{entr} and termination P_{exit} probabilities. Since $P(\mathbf{x}_i|\mathcal{T})$ models not only $\mathcal{T}$ associated observations (true MOVPs) but also those without association (false alarms), the method is able to prune the observations by selecting the most consistent observations, thus forming strong tracks.

3.2.1 Min-cost flow solution

We use the following 0-1 indicators:

$$f_{en,i} = \begin{cases} 1 & \exists T_k \in \mathcal{T},\ T_k \text{ starts from } \mathbf{x}_i \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

$$f_{ex,i} = \begin{cases} 1 & \exists T_k \in \mathcal{T},\ T_k \text{ ends at } \mathbf{x}_i \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

$$f_{i,j} = \begin{cases} 1 & \exists T_k \in \mathcal{T},\ \mathbf{x}_j \text{ is right after } \mathbf{x}_i \text{ in } T_k \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

$$f_i = \begin{cases} 1 & \exists T_k \in \mathcal{T}, \mathbf{x}_i \in T_k \\ 0 & \text{otherwise} \end{cases} \quad (9)$$

and the notations:

$$\begin{aligned} C_{en,i} &= -\log P_{entr}(\mathbf{x}_i) & C_{ex,i} &= -\log P_{exit}(\mathbf{x}_i) \\ C_{i,j} &= -\log P_{link}(\mathbf{x}_j|\mathbf{x}_i) & C_i &= \log \tfrac{\beta_i}{1-\beta_i} \end{aligned} \quad (10)$$

Given the above notations, the objective function (1) in logarithmic form is as follows:

$$\begin{aligned} \mathcal{T}^* = &\ \arg\max_{\mathcal{T}} \textstyle\sum_{T_k \in \mathcal{T}} -\log P(T_k) + \sum_i -\log P(\mathbf{x}_i|\mathcal{T}) \\ = &\ \arg\max_{\mathcal{T}} \textstyle\sum_{T_k \in \mathcal{T}} (C_{en,k_0} f_{en,k_0} \\ &+ \textstyle\sum_j C_{k_j,k_{j+1}} f_{k_j,k_{j+1}} + C_{ex,k_{l_k}} f_{ex,k_{l_k}}) \\ &+ \textstyle\sum_i (-\log(1-\beta_i) f_i - \log \beta_i (1-f_i)) \\ = &\ \arg\max_{\mathcal{T}} \textstyle\sum_i C_{en,i} f_{en,i} + \sum_{i,j} C_{i,j} f_{i,j} \\ &+ \textstyle\sum_i C_{ex,i} f_{ex,i} + \sum_i C_i f_i \end{aligned} \quad (11)$$

subject to that hypotheses in $\mathcal{T}$ do not overlap, equivalent to

$$f_{en,i} + \sum_j f_{j,i} = f_i = f_{ex,i} + \sum_j f_{i,j}, \forall i \quad (12)$$

This still allows for MOVPs to share one vanishing direction, such as the gravity direction, but prohibits tracks in which all vanishing directions are shared.

The MAP formulation, in logarithmic form (11), can now be expressed in terms of a cost-flow network $G(\mathcal{X})$ with source s and sink t, as in [36]. A cost-flow network is depicted in Fig. 2. To each MOVP observation $\mathbf{x}_i \in \mathcal{X}$ correspond two nodes u_i, v_i, an edge (u_i, v_i) of cost $c(u_i, v_i) = C_i$ and flow $f(u_i, v_i) = f_i$, an edge (v_i, t) of cost $c(v_i, t) = C_{ex,i}$ and flow $f(v_i, t) = f_{ex,i}$, and an edge (s, u_i) of cost $c(s, u_i) = C_{en,i}$ and flow $f(s, u_i) = f_{en,i}$. For each $P_{link}(\mathbf{x}_j|\mathbf{x}_i) \neq 0$ will correspond a transition edge

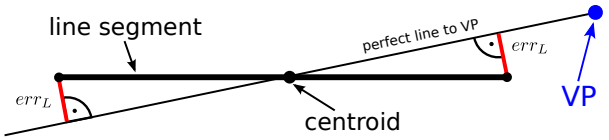

Figure 3: Fitting error for line segment to associated VP: The segment endpoints are projected onto a perfect line from the segment centroid to the VP. The projection error is denoted as err_L.

(v_i, u_j) of cost $c(v_i, u_j) = C_{i,j}$ and flow $f(v_i, u_j) = f_{i,j}$. The eqs. (11) and (12) are equivalent to the flow conservation constraint and the cost of flow in network G. Optimizing over the data association hypothesis $\mathcal{T}^*$ is equivalent to sending the flow from source s to sink t, thus achieving the min-cost flow. To solve for the min-cost flow we use the efficient push-relabel algorithm proposed by Goldberg [10].

3.2.2 Costs

In the following we will define the terms P_{entr}, P_{exit}, P_{link}, and β_i as needed in (10). Entry and exit probabilities are a constant penalty for each started track, similar to a fixed model cost, and can be used to fine-tune the overall sensitivity of the MAP solution. We estimated the best sensitivity to be $P_{entr} = P_{exit} = .01$ on hold-out sequences. The probability P_{link} describes the linking probability for two MOVPs in subsequent frames. Between MOVP $\mathbf{x}_i$ and $\mathbf{x}_j$ we assign a linking probability based on their angular difference[1] α after applying the the camera orientation change $D_{n,n+1}$ computed in § 3.1.

$$P_{link}(\mathbf{x}_j|\mathbf{x}_i) = (1 + e^{\,\gamma_1 \cdot (\alpha - \gamma_2)})^{-1} \quad , \quad (13)$$

where α denotes the angular difference between $D_{n,n+1} \cdot \mathbf{x}_i$ in frame n and $\mathbf{x}_j$ in frame $n+1$. This sigmoid function yields a smooth fall-off at an angular difference of $\alpha - \gamma_2$, with decay rate controlled by γ_1. We learn these parameters on hold out sequences as $\gamma_1 = 4$, and $\gamma_2 = 1$. The remaining probability β_i is the probability of MOVP being a false positive. We set β_i to 1 minus the probability of sampling this MOVP in RANSAC given all detected line segments. To achieve this, we set β_i to 1 minus the percentage of all MOVP samples created in the RANSAC candidate generation step (§ 3.1) which agree with the MOVP candidate i. To be robust against missed line detection we set the limit $C_i = \min(C_i, 0)$.

3.3. MOVP refinement

From the data association in § 3.2 we obtain tracks T_k which contain linked MOVP observations $\mathbf{x}_i$. Each $\mathbf{x}_i$ defines one MOVP as 3×3 orthonormal matrix $\in$ SO(3)

[1] For all angular differences between MOVPs we follow [13], but take care to consider that axes may be ordered differently between MOVPs.

within the local reference frame of the camera. With respect to the global reference frame, all MOVP observations $\mathbf{x}_i$ in each track have to be constant. Using the hypotheses for camera orientation change $D_{n,n+1}, \forall n \in [1, N-1]$ between all frames, we can transform all MOVPs to the global camera reference frame.

We initialize the global camera orientation as $R_1 = diag([1\ 1\ 1])$ for the first frame and $R_n = D_{n-1} \cdot R_{n-1}$ for subsequent frames. We set $\mathcal{R} = \{R_1, \ldots, R_N\}$. For each track T_k, starting at frame S_k, and all observations $\mathbf{x}_i$ we initialize a global MOVP M_k by transforming all observations to the global reference frame and averaging the SO(3) matrices as unit quaternions:

$$M_k = |T_k|^{-1} \sum_i Q\left(R^T_{i+S_k-1} \cdot x_i\right) \quad , \qquad (14)$$

where Q computes the quaternions for a SO(3) matrix. We normalize M_k to unit norm, and convert the quaternions to an SO(3) matrix. We set $\mathcal{M} = \{M_1, \ldots, M_K\}$

Because of the accumulation of errors in $\mathcal{R}$, and noise in frame-wise extracted $\mathbf{x}_i$ the solution for global orientation $\mathcal{R}$ and the discovered MOVPs $\mathcal{M}$ will generally not fit the line segments in each frame perfectly. We refine the initial solution by jointly optimizing $\mathcal{R}$ and $\mathcal{M}$ for the fitting errors of all line segments to all associated MOVPs in all frames in a Non-Linear Least-Squares framework:

$$RSS(\mathcal{R}, \mathcal{M}) = \sum_k^K \sum_i^{|T_k|} err_L(R_{i+S_k-1} \cdot M_k, L_{k,i})$$

The error function err_L accepts a MOVP defined in a camera-centric reference frame and line segments $L_{k,i}$ associated to each vanishing direction. The line segment consistency error is computed for each line segment by projecting the segment endpoints onto a hypothesized perfect line through the line segment centroid and the associated VP. Fig. 3 illustrates this. Using this projection error has the advantage of using the undistorted image-space MOVP fitting error, treating finite and infinite VPs uniformly, and explicitly giving more weights to longer segments [26]. Since the problem is very sparse the optimization is tractable even for long sequences using a Trust-Region minimization, as often used in similar Bundle Adjustment problems [31]. After jointly minimizing the squared endpoint errors for all MOVPs in all frames we obtain optimal camera orientation estimates $\mathcal{R}$, and MOVPs $\mathcal{M}$.

4. Experiments

We conducted two experiments. First, we evaluated our approach on a new dataset of 6 inner-city sequences, each 100 frames long, using established Multi-Object tracking metrics. Second, we evaluated how reliably we can extract the global camera pose over a large dataset of street-view videos provided by a recent video registration work [16].

Figure 4: Example frames of the new dataset used in § 4.1 for sequences 1 to 6 (top left to bottom right)

4.1. MOVPs discovery

Benchmark. We evaluate with three metrics commonly used in tracking: *multi-object tracking accuracy* (MOTA, higher=better), *multi-object tracking precision*, the angular matching error, (MOTP, lower=better) [5], and *ID Switches* (ids, lower=better) [18]. The VP matching threshold was set to an angle of 5 degrees. We collected 6 sequences, each with 100 frames, and manually annotated sets of MOVPs in every 10th frame. Example frames for all videos are shown in Fig. 4. We included MOVP identity information over time. In each sequence between 1 and 4 MOVPs are jointly visible. The videos and annotations will be made public.

Methods. We evaluated our approach on these videos and visualize qualitative results for several frames of one sequence in Fig. 7. In the quantitative evaluation we compare four methods: 1) Our method including optimal tracking of § 3.2 and refinement of § 3.3, 2) our method without refinement, against 3) greedy MOVP association with refinement, and 4) greedy association without refinement.

For the greedy association we start from the same MOVP candidates as described in § 3.1. Instead of optimal data association using min-cost flow algorithm we greedily grow MOVP tracks. Initially, the set of MOVP tracks is empty. For a new frame we merge MOVP observations to existing MOVP tracks if the angular difference is smaller then α degrees. Since the performance of the greedy tracking is strongly dependent on α, we evaluated with multiple values for α and compared to the best result with $\alpha = 6$. The remaining MOVPs of this new frame start new tracks. We remove MOVP tracks shorter than 5 frames. In Fig. 5 we provide a qualitative evaluation.

Results. Adding a refinement step generally improves the greedy as well as the optimal tracking. The benefit of Least-Squares refinement of line endpoint errors is most

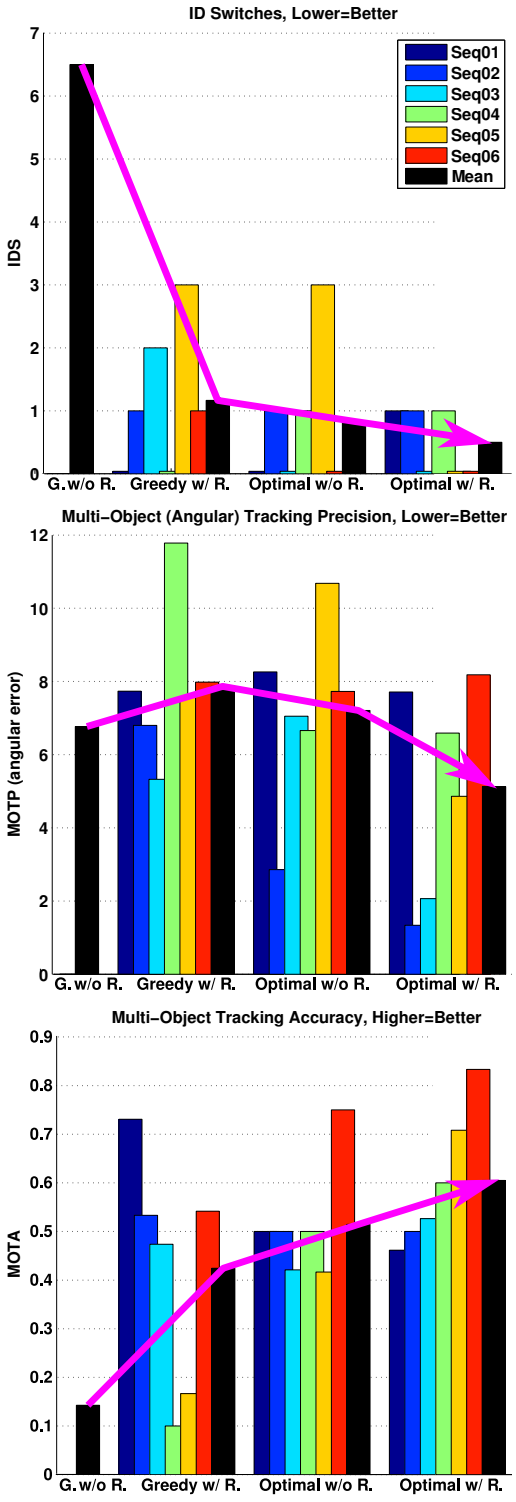

Figure 5: ID Switches, MOTP, MOTA for MOVP discovery in 6 sequences. We compare: 1) Greedy tracking without refinement (only mean is displayed), 2) Greedy tracking with refinement, 3) Optimal MOVP tracking without refinement, 4) optimal MOVP tracking with refinement. Adding refinement generally improves greedy and optimal tracking. Our method outperforms the greedy tracking in all metrics.

visible in sequences in which many MOVPs are visible simultaneously. This is because in those cases, each MOVP may not be very strong or reliable, and the camera orientation change hypotheses $D_{n,n+1}$ may be noisy as well. Especially in these cases enforcing a joint agreement on a global orientation and static MOVPs in the global frame improves results. We also observe, that our optimal tracking outperforms the refined greedy tracking even when no refinement is employed. Errors influencing the MOTA scores for the greedy and optimal tracking are largely dominated by false positive tracks, which may share strong line support with other MOVPs, such as on the gravity direction. Since they have partial strong line support and may move consistently with other MOVPs sometimes they are incorrectly included in the tracks. The greedy and optimal tracking both suffer equally from this problem.

It is important to emphasize that MOTA, MOTP and ID Switches are strongly interdependent, and that no singular focus on a single metric should be placed. It is possible to achieve good MOTP, i.e. low angular error, as for the greedy tracking without refinement, by simply accepting the closest tracks in each frame regardless of temporal consistency, which results in many ID Switches. Conversely, similar MOTA scores over all methods, as for sequence 3, are only a good discriminative metric, if information about the ID Switches on ground truth tracks is considered as well.

Runtimes. The greedy and optimal tracking, both with refinement, run for 9.6 and 9.8 seconds per frame, respectively. The runtimes are largely dominated by our unoptimized MATLAB implementation of MOVP candidate generation, which runs for 9.1 seconds per frame on average.

4.2. Camera orientation estimation

Benchmark. In the first experiment we evaluated how accurate we can discover all MOVPs in the scene. For many tasks the identification of Manhattan Frames (MOVPs) is just the first step in discovering a more fine-grained scene structure. Manhattan frame discovery can help in this, since we get the camera orientation change for free when at least two VPs are identified in two different views [6]. After the refinement, proposed in § 3.3, we obtain a global camera orientation estimate, which we will evaluate in this section.

The Antwerp Street-View Dataset, introduced in [16] and used for Video Registration, provides 48 sequences of 301 frames with precisely known camera pose at all times. Several example frames are shown in Fig. 1. In order to make the orientation estimation more challenging we uniformly subsampled the sequences to 101 frames.

Methods. We track multiple MOVPs in all 48 sequences using the greedy and optimal MOVP discovery, including refinement for both methods. We also compared to the hypothesized global orientation before refinement, as mentioned in § 3.3. Additionally, we extracted SIFT features,

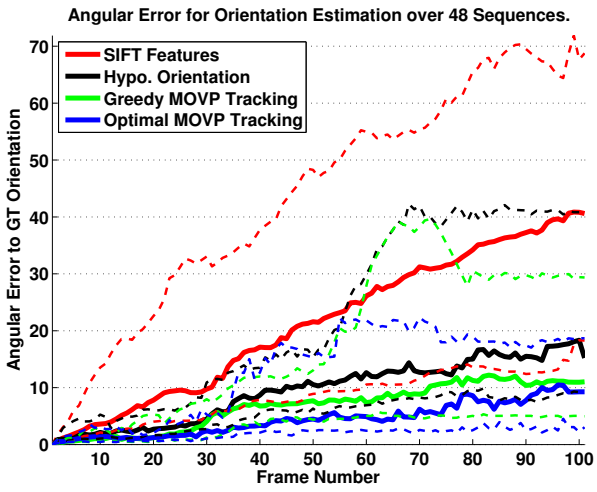

Figure 6: Error accumulation of global camera orientation estimation on the Antwerp Street-View dataset [16] for our optimal MOVP tracking, greedy MOVP tracking, hypothesized global orientations before refinement, and frame-to-frame SIFT features matching with essential matrix decomposition. The dotted lines in each color denote the 75% and 25% quantiles for each method.

computed Essential matrices between successive frames, extracted the frame-to-frame change in camera orientation, and transformed it into a global camera orientation estimate, as we did in § 3.3 for the hypothesized camera orientation.

Results. The comparison of all four methods is plotted in Fig. 6. We notice that already the hypothesized camera orientation, even before refinement, has half the accumulated orientation error of SIFT features. We again half this drift error by adding greedy or optimal MOVP tracking and refinement. The optimal tracking gains over the greedy tracking mostly in the worst case scenarios, where fewer mistakes result in fewer local optima in which the refinement can get trapped. Overall optimal MOVP tracking and refinement results in less than 10 degree orientation drift (on median) against 40 degrees for SIFT features.

5. Conclusion

In this work we presented a novel method for discovery of sets of mutually orthogonal vanishing points from monocular video sequences with unknown camera pose. We contribute an optimal way of extracting MOVPs over time using a hypothesized global orientation from all MOVP candidates, and a method to jointly refine MOVPs and global camera poses. This refinement, similar in spirit to Bundle Adjustment for Structure-from-Motion problems, greatly improves both greedy and optimal MOVP tracking results. Since we are the first to tackle this problem, we introduce a new dataset for MOVP discovery, and will make the videos and MOVP annotation publicly available. [2]

In future work we plan to tackle current limitations of the method: 1) false positives due to strong shared VPs and line association ambiguity for VPs on the horizon line. 2) Our method is generic and does not favor specific VPs. However, when considering city scenes, detecting zenith and horizon lines could provide powerful additional constraints.

Acknowledgments: This work was partly supported by the European Research Council (ERC) under the project VarCity (#273940) and by the ETH General Founding (OK).

References

[1] M. E. Antone and S. Teller. Automatic Recovery of Relative Camera Rotations for Urban Scenes. *CVPR*, 2000. 1, 2

[2] M. Antunes and J. a. P. Barreto. A Global Approach for the Detection of Vanishing Points and Mutually Orthogonal Vanishing Directions. *CVPR*, 2013. 2

[3] S. T. Barnard. Interpreting Perspective Images. *Artificial Intelligence*, 1982. 2

[4] J.-C. Bazin and M. Pollefeys. 3-line RANSAC for Orthogonal Vanishing Point Detection. *IROS*, 2012. 2, 3

[5] K. Bernardin, A. Elbs, and R. Stiefelhagen. Multiple Object Tracking Performance Metrics and Evaluation in a Smart Room Environment Performance Metrics for Multiple Object Tracking. *EURASIP*, 2008. 5

[6] B. Caprile and V. Torre. Using Vanishing Points for Camera Calibration. *IJCV*, 1990. 2, 6

[7] D. J. Crandall, A. Owens, N. Snavely, and D. P. Huttenlocher. SfM with MRFs: Discrete-Continuous Optimization for Large-Scale Structure from Motion. *PAMI*, 2012. 1

[8] P. Denis, J. H. Elder, and F. J. Estrada. Efficient Edge-Based Methods for Estimating Manhattan Frames in Urban Imagery. *ECCV*, 2008. 2

[9] W. Elloumi, S. Treuillet, and R. Leconge. Tracking Orthogonal Vanishing Points in Video Sequences for a Reliable Camera Orientation in Manhattan World. *CISP*, 2012. 1, 2

[10] A. V. Goldberg. An efficient implementation of a scaling minimum-cost flow algorithm. *J. Algorithms*, 1997. 4

[11] L. Grammatikopoulos, G. Karras, and E. Petsa. An Automatic Approach for Camera Calibration from Vanishing Points. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2007. 1, 2

[12] M. Hornáček and S. Maierhofer. Extracting Vanishing Points across Multiple Views. *CVPR*, 2011. 1, 2

[13] D. Q. Huynh. Metrics for 3D Rotations: Comparison and Analysis. *Journal of Mathematical Imaging and Vision*, 35(2):155–164, June 2009. 4

[14] C. Kim and R. Manduchi. Planar Structures from Line Correspondences in a Manhattan World. *ACCV*, 2014. 1

[15] J. Košecká and W. Zhang. Video Compass. *ECCV*, 2002. 2

[16] T. Kroeger and L. Van Gool. Video Registration to SfM Models. *ECCV*, 2014. 2, 5, 6, 7

[2] http://www.vision.ee.ethz.ch/~kroegert/

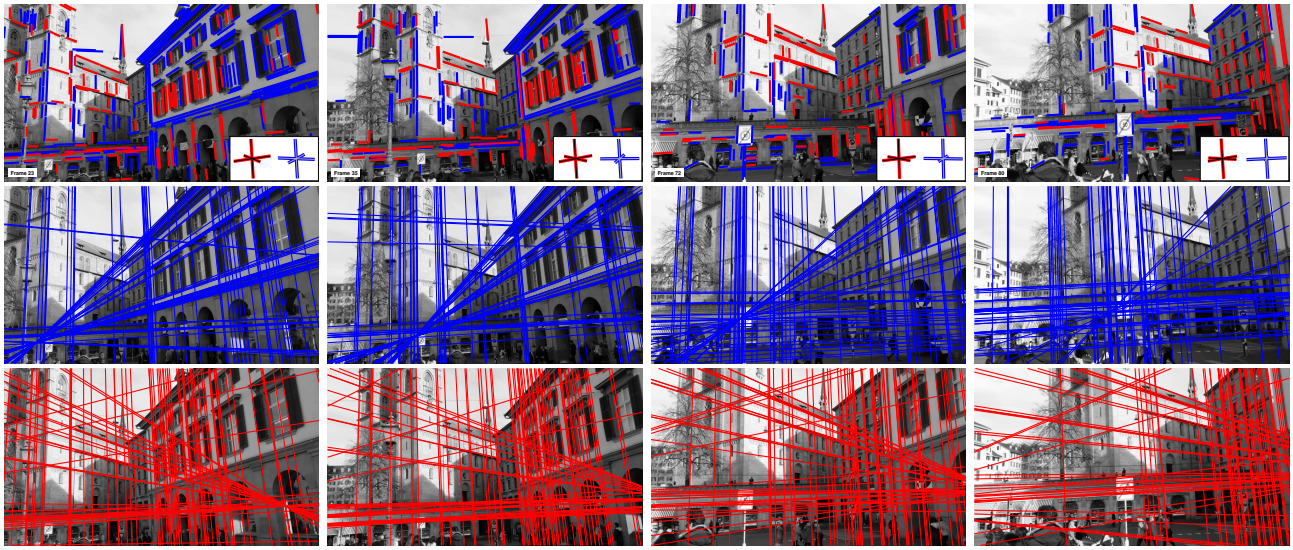

Figure 7: Qualitative results two discovered MOVPs in one sequence using our method. Top: Four frames of the sequence with line segments, colored according to MOVP assignment. Bottom two rows: All line segments are extended to the point of convergence. The two tracks share a common vertical VP corresponding to the gravity direction.

[17] J. Lezama, R. G. Von Gioi, G. Randall, and J.-m. Morel. Finding Vanishing Points via Point Alignments in Image Primal and Dual Domains. *CVPR*, 2014. 2

[18] Y. Li, C. Huang, and R. Nevatia. Learning to associate: HybridBoosted multi-target tracker for crowded scene. *CVPR*, 2009. 5

[19] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *IJCV*, 2004. 3

[20] E. Lutton, H. Maître, and J. Lodez-Krahe. Contribution to the Determination of Vanishing Points Using Hough Transform. *PAMI*, 1994. 2

[21] M. J. Magee and J. K. Aggarwal. Determining vanishing points from perspective images. *Computer Vision, Graphics, and Image Processing*, 1984. 2

[22] B. Micusik and H. Wildenauer. Minimal solution for uncalibrated absolute pose problem with a known vanishing point. *3DVV*, 2013. 1, 2

[23] P. Moghadam and J. F. Dong. Road Direction Detection Based on Vanishing-Point Tracking. *IROS*, 2012. 1, 2

[24] L. Quan and R. Mohr. Determining perspective structures using hierarchical Hough transform. *Pattern Recognition Letters*, 1989. 2

[25] C. Rasmussen. RoadCompass: following rural roads with vision + ladar using vanishing point tracking. *Autonomous Robots*, 2008. 1, 2

[26] C. Rother. A new approach to vanishing point detection in architectural environments. *Image and Vision Computing*, 20, 2002. 2, 5

[27] G. Schindler and F. Dellaert. Atlanta World: An Expectation Maximization Framework for Simultaneous Low-level Edge Grouping and Camera Calibration in Complex Man-made Environments. *CVPR*, 2004. 2

[28] J. Straub, G. Rosman, O. Freifeld, J. J. Leonard, and J. W. Fisher III. A Mixture of Manhattan Frames : Beyond the Manhattan World. *CVPR*, 2014. 1, 2

[29] J.-P. Tardif. Non-Iterative Approach for Fast and Accurate Vanishing Point Detection. *ICCV*, 2009. 2

[30] E. Tretyak, O. Barinova, P. Kohli, and V. Lempitsky. Geometric Image Parsing in Man-Made Environments. *IJCV*, 2012. 2

[31] B. Triggs, P. F. Mclauchlan, R. I. Hartley, and A. W. Fitzgibbon. Bundle Adjustment A Modern Synthesis. *Proceeding ICCV '99 Proceedings of the International Workshop on Vision Algorithms: Theory and Practice*, 34099:298–372, 2000. 5

[32] T. Tuytelaars, M. Proesmans, and L. Van Gool. The Cascaded Hough Transform as Support for Grouping and Finding Vanishing Points and Lines. In *AFPAC*, 1997. 2

[33] R. G. von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall. LSD: a Line Segment Detector. *IPOL*, 2012. 3

[34] H. Wildenauer and A. Hanbury. Robust Camera Self-Calibration from Monocular Images of Manhattan Worlds. *CVPR*, 2012. 2

[35] Y. Xu, S. Oh, and A. Hoogs. A Minimum Error Vanishing Point Detection Approach for Uncalibrated Monocular Images of Man-made Environments. *CVPR*, 2013. 2

[36] L. Zhang, Y. Li, and R. Nevatia. Global Data Association for Multi-Object Tracking Using Network Flows. *CVPR*, 2008. 3, 4